

PEMBUATAN HAAR-CASCADE DAN LOCAL BINARY PATTERN SEBAGAI SISTEM PENDETEKSI HALANGAN PADA AUTOMATIC GUIDED VEHICLE

Riza Agung Firmansyah

Fakultas Teknologi Industri, Jurusan Teknik Elektro
Institut Teknologi Adhi Tama Surabaya
Email: rizaagungf@itats.ac.id

Enggar Alfianto

Fakultas Teknologi Informasi, Jurusan Sistem Komputer
Institut Teknologi Adhi Tama Surabaya
Email: enggar@itats.ac.id

ABSTRAK

Automatic Guided Vehicle (AGV) merupakan salah satu jenis robot yang bekerja mengikuti suatu jalur. Dalam penelitian ini, AGV digunakan di lingkungan perkantoran kampus. Lingkungan kampus menyebabkan jalur yang dilewati sulit dikondisikan dalam kondisi steril karena banyak objek penghalang. Hal ini membuat robot harus memiliki sistem pendeteksi halangan yang mampu membedakan halangan diam dan bergerak. Dalam penelitian ini penghalang diam berupa bak sampah dan penghalang bergerak adalah manusia. Untuk mendeteksi penghalang tersebut digunakan haar-cascade sebagai pencarian kasar dan local binary pattern (LBP) sebagai pencarian halus. Haar-cascade dibuat dengan memanfaatkan opencv haar training. Training dilakukan dengan menggunakan 300 citra positif dan 2317 citra negative pada masing-masing objek. Haar-cascade classifier didapatkan setelah dilakukan training hingga 10 stage. Haar-cascade diuji pada jarak dibawah 4 meter dari objek. Pencarian halus menggunakan LBP dilakukan saat haar-cascade mendeteksi adanya objek lebih dari satu. Dari pengujian yang telah dilakukan, sistem berhasil mendeteksi adanya halangan dengan tingkat keberhasilan 81,7%.

Kata kunci: automatic guided vehicle; haar-cascade; local binary pattern; sistem pendeteksi halangan.

ABSTRACT

Automatic Guided Vehicle (AGV) is one type of robot that works following a path. In this study, AGV is used in campus or office environments. Campus environment causes the passage path is difficult to be conditioned in sterile conditions because of many obstacles. This makes the robot must have an obstacle detection system that is able to classify between static and moving obstacles. This research use trash bin as static obstacles and human as moving obstacles. To detect the obstacles, this research use haar-cascade as a rough search and local binary pattern as a fine search. Haar-cascade is made by opencv haar training. Training was done using 300 positive images and 2317 negative images on each object. Haar-cascade classifier is obtained after training up to 10 stages. Haar-cascade was tested at a distance below 4 meters from the object. A fine search using a local binary pattern is performed when the haar-cascade detects more than one object. From the tests that have been done, the system managed to detect an obstacle with the success rate of 81,7%

Keywords: automatic guided vehicle; haar-cascade; local binary pattern; obstacle detection.

1. PENDAHULUAN

Automatic Guided Vehicle (AGV) merupakan salah satu aplikasi robot yang digunakan untuk mengangkut benda dari satu tempat ke tempat lain secara otomatis. AGV digunakan pada industri dalam proses manufaktur, perakitan, dan pergudangan [1] dikarenakan efisiensinya tinggi, handal, ketahanan yang kuat, fleksibel, dan akurasi gerak yang tinggi. Sebab itu maka AGV juga merambah ke beberapa bidang. Selain untuk aplikasi industri, AGV juga digunakan pada perkantoran, pergudangan, pertanian, dan rumah sakit [2].

Dalam penelitian ini, AGV dioperasikan di lingkungan perkantoran kampus untuk mengantar barang antar ruangan. Antar ruangan dihubungkan oleh sebuah koridor maupun lorong. Sehingga munculnya objek

penghalang (*obstacle*) di jalur pergerakan robot akan sering terjadi. Penghalang yang dimaksud dalam penelitian ini adalah manusia maupun objek lain seperti tempat sampah, meja atau kursi.

Untuk mencegah robot menabrak sebuah objek maka diperlukan sebuah sistem pendeteksi dan penghindar halangan. Namun dalam penelitian ini lebih difokuskan untuk mendeteksi jenis penghalang. Dengan diketahuinya jenis penghalang maka dapat diketahui pula penghalang tersebut bertipe diam (*static obstacle*) atau bergerak (*moving obstacle*). Hal ini cukup penting mengingat sistem penghindar halangan dipengaruhi pula oleh tipe penghalang. Penghalang yang dideteksi dalam penelitian ini adalah manusia sebagai *moving obstacle* dan tempat/bak sampah sebagai *static obstacle*. Untuk mendeteksi kedua penghalang tersebut maka diperlukan teknik pengolahan citra.

Pengolahan citra perlu digunakan sebab beberapa sistem deteksi halangan yang menggunakan sensor *ultrasonic* [3] tidak mampu mengetahui jenis penghalang. Penggunaan *ultrasonic* dan *infrared* [4] hanya mampu mengetahui adanya penghalang statis. Penelitian [5] mampu mendeteksi material objek namun belum diuji pada objek manusia. Dengan menggunakan teknik pengolahan citra diharapkan jenis penghalang dapat diketahui. Salah satu teknik pengolahan citra yang bisa digunakan adalah *haar-cascade*.

Haar-cascade biasa digunakan untuk mendeteksi objek seperti mobil [6], pejalan kaki [7] dan objek lainnya. Namun penggunaannya jarang digunakan pada mobile robot. Sehingga dalam penelitian ini akan dicoba dalam aplikasi mobile robot. Selain *haar-cascade*, teknik pengolahan citra seperti deteksi warna [8] [9], deteksi kontur [10] dan *extended histogram oriented gradient* (EHOG) [6] telah diimplementasikan oleh beberapa robot. Namun penggunaan deteksi warna dan kontur tidak mampu mengenali jenis penghalang karena bekerja dengan membedakan warna objek. Sedangkan penggunaan EHOG cukup baik namun sangat dipengaruhi iluminasi cahaya.

Penggunaan *haar-cascade* cukup baik dalam mendeteksi objek. Namun dalam kondisi tertentu *haar-cascade* mendeteksi objek palsu sehingga akan memunculkan beberapa kandidat objek. Objek palsu ini biasanya berupa objek yang pola atau bentuknya hampir sama dengan objek yang akan dideteksi. Untuk mengatasi hal tersebut maka dalam penelitian ini diimplementasikan *local binary pattern* (LBP) sebagai pencarian halus (*fine search*). Hal ini disebabkan LBP biasa digunakan untuk pattern recognition [11]. Penelitian ini masih dalam tahap pengujian *haar-cascade* dan *local binary pattern* sehingga pengujian belum diimplementasikan pada robot secara langsung.

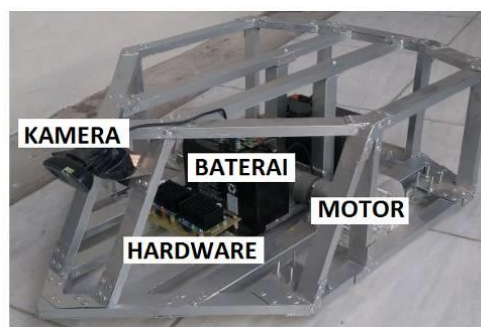
2. METODOLOGI PENELITIAN

Penelitian ini dilakukan dalam beberapa tahap. Tahapan tersebut antara lain pengumpulan citra sampel, proses training, pengujian deteksi objek dengan *haar-cascade*, dan pengujian *fine search* menggunakan *local binary pattern*. Proses pembuatan mekanik robot dan perangkat keras tidak dilakukan dalam penelitian ini. Hal ini dikarenakan penelitian ini menggunakan robot yang sudah ada sebelumnya.

2.1 Automatic Guided Vehicle yang Digunakan

Automatic Guided Vehicle (AGV) sebagian besar digunakan di dalam sistem manufaktur untuk bongkar muat material. Penggunaan utama AGV adalah untuk mengirimkan barang dari satu lokasi ke lokasi lain dengan lebih mudah [12]. Saat bergerak, AGV pada umumnya mengikuti pemandu (*guide*) berupa garis, *magnetic tape*, RFID, dan beberapa jenis lainnya.

Dalam penelitian ini, AGV berada di lingkungan perkantoran/kampus sehingga penggunaan pemandu berupa garis dan sejenisnya harus dihindari. Jika hal ini dilakukan maka akan mengurangi kebersihan dan kerapian lingkungan. Agar AGV mampu digunakan dalam area perkantoran tanpa memasang garis pada lantai, jalur pemandu yang digunakan harus merupakan benda asli yang berada di area tersebut. Jalur pemandu yang digunakan bisa berupa dinding ruangan, pagar, dan lain-lain.



Gambar 1. AGV Yang Digunakan Dalam Penelitian

AGV yang digunakan menggunakan tiga buah sensor navigasi. Sensor tersebut antara lain kamera, *ultrasonic rangefinder*, dan kompas digital. Sensor *ultrasonic* digunakan untuk mendeteksi dinding yang selanjutnya digunakan sebagai pemandu (*guide*). Sensor kompas digital digunakan untuk membaca arah hadap AGV. Sedangkan kamera digunakan untuk mendeteksi penghalang. Namun dalam penelitian ini lebih difokuskan pada sensor kamera untuk mendeteksi penghalang. AGV yang digunakan ditunjukkan pada Gambar 1.

2.2 Obstacle Detection Pada Mobile Robot

Untuk melakukan deteksi halangan dapat digunakan sensor *ultrasonic* [3]. Deteksi halangan dilakukan dengan melakukan pembacaan jarak menggunakan sensor *ultrasonic*. Jika jarak yang terbaca kurang dari nilai batas maka sistem mendeteksi adanya objek penghalang. Cara yang sama juga dilakukan [4] namun sensor yang digunakan adalah *infrared*.

Ultrasonic dan *infrared* yang umumnya digunakan sebagai pengukur jarak dapat digunakan untuk mengetahui jenis material [5]. Material yang berhasil dikenali adalah kertas, plastik, kayu, dan beberapa objek lain. Pengenalan jenis material dilakukan dengan membandingkan data pembacaan jarak antara *ultrasonic* dan *infrared*. Namun pengenalan belum diuji pada objek manusia.

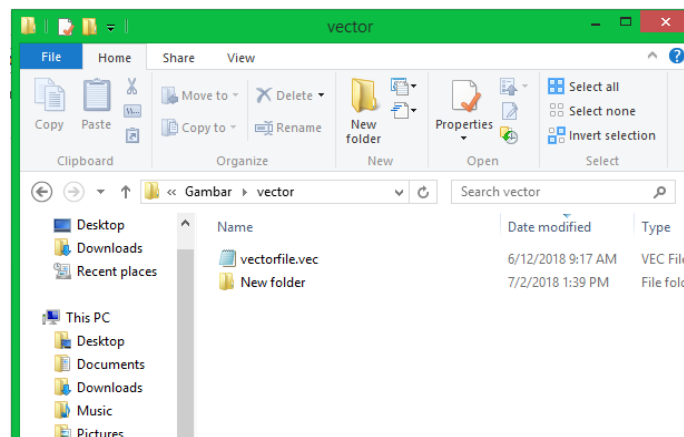


Gambar 2. Deteksi Penghalang Berdasarkan Warna [8]

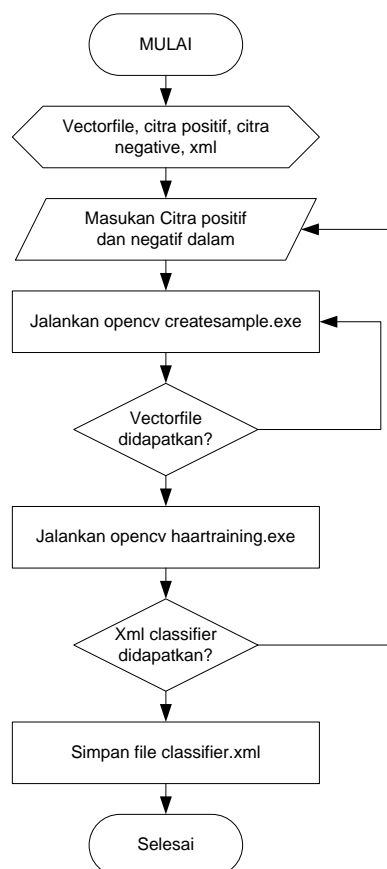
Sistem deteksi dan penghindar halangan [8] menggunakan deteksi warna untuk menentukan posisi penghalang. Penghalang yang digunakan berupa tabung berwarna kuning yang diletakkan secara acak di sekitar robot. Unit pemroses yang digunakan adalah Raspberry PI yang dihubungkan dengan arduino uno dengan komunikasi serial. Kamera yang digunakan adalah kamera webcam WB100 dengan resolusi 5 MP yang hasilnya dapat dilihat pada gambar 2. Penggunaan deteksi warna cukup mudah dilakukan saat penghalang berwarna sama. Namun belum bisa digunakan jika penghalang berbeda warna.

Deteksi penghalang yang lain adalah dengan menggunakan metode segmentasi warna [9]. Citra yang telah diperoleh selanjutnya diolah berdasarkan model warna HSV. Setelah dilakukan proses *threshold*, deteksi kontur, dan ekstraksi fitur selanjutnya sistem mengirimkan koordinat objek penghalang menggunakan komunikasi UART.

Deteksi objek penghalang bisa juga dengan mendeteksi kontur objeknya kontur [10]. Deteksi kontur tersebut dilakukan dengan meletakkan kamera di atas area kerja. Kamera tersebut mengambil citra robot dan penghalangnya dari atas. Cara ini menghasilkan hasil yang cukup bagus namun tidak bisa diterapkan dalam penelitian ini. Hal tersebut dikarenakan area kerja robot luas sehingga tidak mungkin terjangkau seluruhnya oleh kamera.



Gambar 3. Hasil Create Sample Yang Berupa Vectorfile



Gambar 4. Diagram Alir Proses Haar-Cascade

2.3 Haar-Cascade

Haar-cascade merupakan sebuah *classifier* (penggolong) yang digunakan untuk mendeteksi objek yang telah dilatih sebelumnya. *Haar-cascade* yang dibuat pertama dalam penelitian ini adalah untuk mendeteksi tempat/bak sampah. Proses diawali dengan mengumpulkan citra tempat/bak sampah sebagai citra positif sebanyak 300 buah dan citra *negative* sebanyak 2317 buah. Setelah seluruh citra terkumpul maka dibuat vector dari citra tersebut menggunakan aplikasi opencv createsamples.exe. Hasil proses tersebut berupa file dengan ekstensi *.vec yang ditunjukkan pada gambar 3 sedangkan diagram alir proses training haar-cascade ditunjukkan oleh Gambar 4.

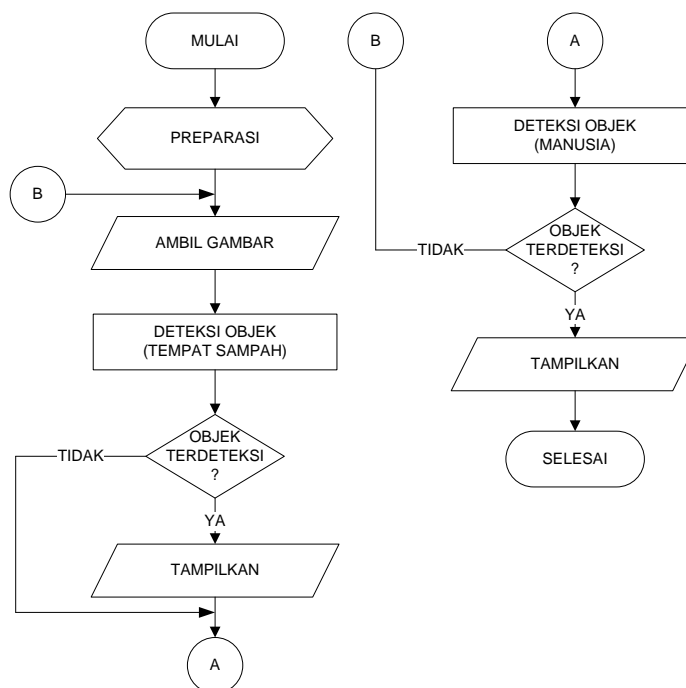
```

C:\Windows\system32\cmd.exe

Parent node: 0
*** 1 cluster ***
POS: 20 20 1.000000
NEG: 624 0.0051419
BACKGROUND PROCESSING TIME: 1.04
Precalculation time: 5.23
+-----+
| N | %SMP | F | ST.THR | HR | FA | EXP. ERR |
+-----+
| 1 | 100% | - | 0.993635 | 1.000000 | 0.858974 | 0.832298 |
+-----+
| 2 | 86%  | + | 0.085168 | 1.000000 | 0.858974 | 0.832298 |
+-----+
| 3 | 83%  | - | 1.085167 | 1.000000 | 0.852564 | 0.863354 |
+-----+
| 4 | 82%  | + | 1.085167 | 1.000000 | 0.852564 | 0.846273 |
+-----+
| 5 | 83%  | - | 2.085167 | 1.000000 | 0.001603 | 0.849379 |
+-----+
Stage training time: 4.53
Number of used features: 5
Parent node: 0
  
```

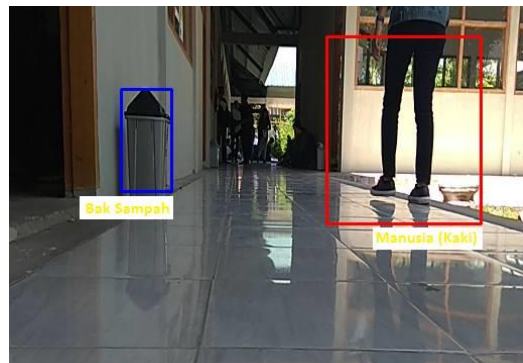
Gambar 5. Proses Haartraining pada Stage Satu

Setelah citra positif dan negative disiapkan langkah selanjutnya melakukan proses training. Proses training dilakukan dengan menggunakan aplikasi opencv haartraining.exe. Saat proses training dijalankan maka komputer akan menunjukkan tampilan seperti pada gambar 5. Setelah training selesai dilakukan maka akan didapatkan file cascade classifier dengan ekstensi *.xml. Proses yang sama juga dilakukan untuk mendeteksi penghalang berupa manusia yang dalam hal ini dilakukan dengan mendeteksi kaki.



Gambar 6. Diagram Alir Proses Deteksi Objek (Penghalang)

Algoritma deteksi objek nantinya akan diimplementasikan ke dalam raspberry PI dengan memanfaatkan file *cascade-classifier* yang didapatkan. Objek yang akan dideteksi ada dua jenis sehingga proses deteksi dilakukan dalam dua tahap. Pertama adalah mendeteksi keberadaan tempat/bak sampah. Dan kedua adalah mendeteksi keberadaan manusia yang berada di lorong/koridor. Proses ini diilustrasikan oleh diagram alir gambar 6.



Gambar 7. Hasil Deteksi Sistem Yang Mendeteksi Bak Sampah Dan Kaki

Proses diawali dengan mengambil sebuah sampel gambar dari pembacaan kamera. Selanjutnya menggunakan deteksi objek menggunakan *haar-cascade* untuk mencari tempat/bak sampah. Jika ditemukan, selanjutnya posisi tempat/bak sampah ditandai dengan persegi panjang berwarna biru. Namun jika tidak ditemukan maka langsung dilakukan proses deteksi manusia menggunakan *haar-cascade*. Hasil deteksi tempat/bak sampah dan manusia dapat dilihat pada gambar 7. Object detection yang dilakukan menggunakan *haar-cascade* kadang memunculkan lebih dari satu kandidat objek. Hal ini dapat disebabkan adanya kemiripan pola, warna, atau faktor lainnya. Sebagai contoh, kondisi ini ditunjukkan pada Gambar 8.



Gambar 8. Kondisi Saat Detector Mengenali Objek Palsu / Menyerupai

2.4 Local Binary Pattern

Local binary pattern (LBP) memiliki performa yang sangat baik saat digunakan pada klasifikasi tekstur dan inspeksi permukaan. LBP dapat didefinisikan sebagai perbandingan nilai intensitas piksel pusat dengan nilai *neighborhood pixel*. Sebagai contoh pada citra berukuran 3x3, nilai intensitas pada pusat citra dibandingkan dengan *neighborhood pixel*. Dengan cara membandingkan nilai piksel pusat citra dengan nilai *neighborhood pixel*, jika piksel pusat lebih besar maka diberi nilai 1 dan jika piksel pusat lebih kecil maka diberi nilai 0. Untuk mendapatkan nilai LBP dari citra grayscale dapat digunakan persamaan (1) dan (2). Proses konversi citra grayscale menjadi LBP diilustrasikan pada Gambar 9.

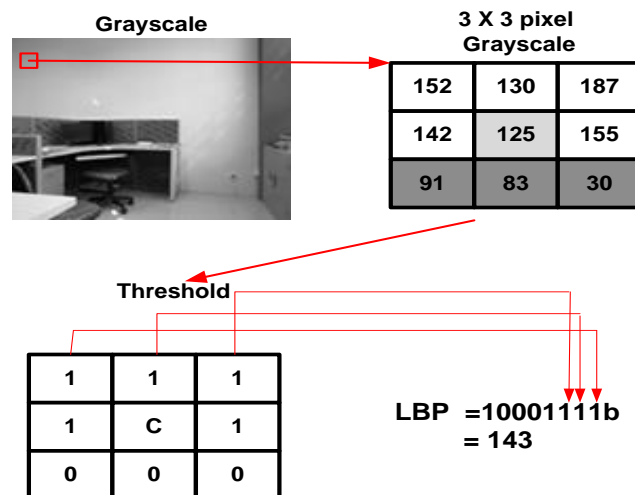
$$s(g_0, g_i) = \begin{cases} 0, & g_i < g_0 \\ 1, & g_i \geq g_0 \end{cases} \quad 1 \leq i \leq 8 \quad (1)$$

$$LBP(g_0) = s(g_0, g_i) \cdot 2^{i-1} \quad (2)$$

Kemiripan suatu pola pada citra LBP dapat diketahui berdasarkan histogram yang didapatkan. Histogram adalah representasi grafis untuk distribusi warna dari citra digital. Histogram tersebut dapat digunakan untuk mewakili suatu citra tertentu. Sehingga untuk mencari kemiripan suatu gambar dapat dilakukan dengan mencocokkan histogramnya. *Histogram matching* adalah salah satu metode dalam pengolahan citra digital yang berfungsi untuk mengetahui kesamaan atau kemiripan antara dua histogram atau lebih. *Histogram matching* pada umumnya dapat dilakukan dengan menggunakan persamaan *histogram intersection* yang ditunjukkan (3). Jika H1 dan H2 memiliki kesesuaian yang besar, maka nilai x2 akan bernilai mendekati nol.

$$x2(H1, H2) = \frac{1}{2} \sum_i \frac{(H1_i - H2_i)^2}{(H1_i + H2_i)} \quad (3)$$

Dimana, $x2$ adalah Nilai kesamaan histogram
 H_1 adalah Histogram yang dijadikan acuan
 H_2 adalah Histogram yang akan dicari kesamaanya

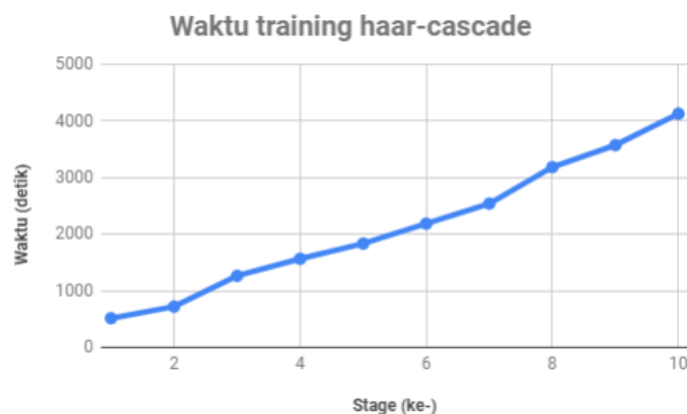


Gambar 9. Ilustrasi Konversi *Grayscale* Menjadi LBP

3. HASIL DAN PEMBAHASAN

3.1 *Proses Training Menggunakan Opencv Haartraining*

Proses training dilakukan dengan menggunakan 300 citra positif dan 2317 citra negatif pada tiap objek yang akan di deteksi. Prosedur yang dilakukan seperti yang telah dijelaskan di bab II.C. Dalam penelitian ini training dilakukan sebanyak 10 *stage*. Semakin tinggi *stage* hasil deteksi menjadi lebih akurat namun memerlukan waktu semakin lama. Proses training ini membutuhkan waktu hingga lebih tujuh jam. Gambar 10 menunjukkan waktu yang dibutuhkan haartraining pada tiap *stage*.



Gambar 10. Waktu Yang Dibutuhkan *Haartraining* pada Tiap *Stage*

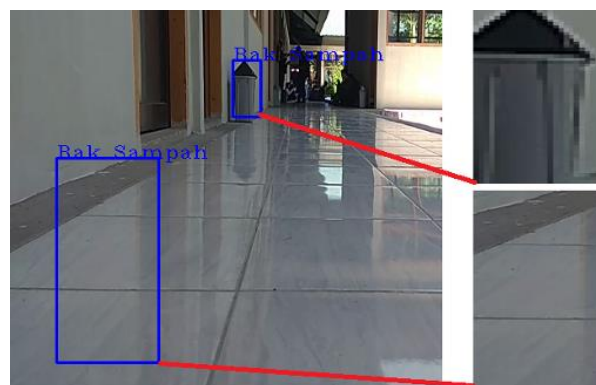
3.2 *Hasil Pengujian Deteksi Halangan Menggunakan Haar-Cascade*

Pengujian pertama dilakukan untuk menguji tingkat keberhasilan *haar-cascade* dalam mendeteksi penghalang. Pengujian ini dilakukan dengan menaruh objek penghalang pada koridor. Selanjutnya robot digerakan pada jarak 6 meter hingga mendekati objek penghalang. Pada proses tersebut, kamera mengambil gambar dan dilanjutkan dengan proses deteksi. Gambar yang diambil sebanyak 1000 buah dan dihitung jumlah keberhasilan sistem dalam mendeteksi objek.



Gambar 11. Sistem Berhasil Mendeteksi Adanya Tempat/Bak Sampah

Sistem dianggap berhasil jika mampu mendeteksi penghalang dengan tepat, baik secara posisi, jenis penghalang, maupun jumlah penghalang. Sebagai contoh gambar 7 dan gambar 11 merupakan hasil keberhasilan sistem mendeteksi penghalang. Sedangkan pada gambar 12 dianggap gagal karena tidak memenuhi kriteria keberhasilan. Dari pengujian yang dilakukan, sistem berhasil mendeteksi objek sebanyak 786 kali. Dengan hasil tersebut *haar-cascade* mampu mendeteksi objek dengan tingkat keberhasilan 78,6 %.

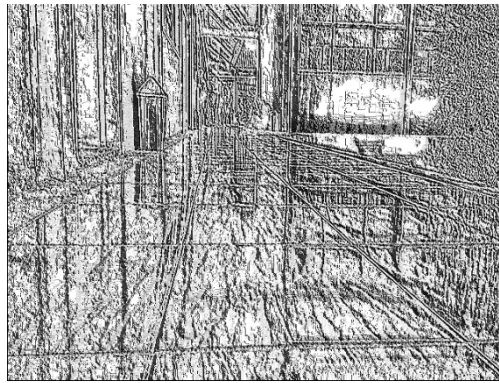


Gambar 12. Kegagalan Sistem Mendeteksi Posisi Dan Jumlah Objek Dengan Benar

Pada gambar 12 dapat dilihat bahwa sistem menemukan 2 objek bak sampah. Namun sebenarnya hanya ada satu bak sampah yang berada pada area jangkauan kamera. Objek lain bisa dikatakan sebagai objek palsu. Dari pengamatan sekilas, terlihat bahwa objek asli dan palsu memiliki kesamaan pola. Pola tersebut berada di bagian atas yang lebih gelap dibanding area bawah. Untuk memilih objek mana yang merupakan objek asli diperlukan sebuah pencarian halus.

3.3 Hasil Pengujian Deteksi Halangan dengan Penambahan LBP

Hasil deteksi menggunakan *haar-cascade* menunjukkan nilai yang cukup bagus. Namun untuk meningkatkan keberhasilan sistem, proses pencarian halus menggunakan *local binary pattern* (LBP) perlu dilakukan. Proses ini hanya dilakukan saat *haar-cascade* mendeteksi lebih dari satu objek (objek palsu). Prosedur yang dilakukan sama dengan pengujian pertama. Dari 1000 sampel gambar, terdapat 38 gambar yang terdeteksi ada objek palsu. Sehingga proses pencarian halus hanya dilakukan pada 38 gambar tersebut. Salah satu gambar yang mendeteksi adanya objek palsu ditunjukkan oleh gambar 12.



Gambar 13. Hasil Konversi Citra RGB Menjadi LBP

Proses pencarian halus dilakukan dengan mengkonversi citra objek dan kandidatnya menjadi citra LBP. Namun sebelumnya citra tersebut diubah ukurannya menjadi 24x48 piksel. Citra LBP selanjutnya dicari histogramnya dan dilakukan proses *histogram matching* menggunakan persamaan 3. Kandidat yang memiliki nilai *histogram matching* paling kecil dianggap sebagai objek yang dicari. Hasil gambar 11 saat dikonversi menjadi LBP ditunjukkan pada gambar 13. Sedangkan hasil akhir pencarian halus menggunakan LBP ditunjukkan oleh gambar 14. Dari 38 gambar yang diproses, sistem berhasil mendapatkan 31 hasil yang sesuai sehingga total gambar yang berhasil ditemukan adanya objek sebanyak 817.



Gambar 14. Hasil Akhir Pencarian Halus Menggunakan LBP

4. KESIMPULAN

Berdasarkan pengujian yang telah dilakukan, sistem pendeteksi halangan mampu mendeteksi adanya objek penghalang beserta jenisnya. Sehingga sistem ini dapat dijadikan masukan dari sistem penghindar halangan yang mampu menghindari penghalang statis maupun dinamis. Dari 1000 pengujian, kombinasi *haar-cascade* sebagai pencarian kasar dan *local binary pattern* sebagai pencarian halus berhasil mengenali objek sebanyak 817 kali atau 81.7 %.

UCAPAN TERIMA KASIH

Penelitian ini merupakan luaran dalam penelitian dosen pemula 2018. Sehingga dalam kesempatan ini penulis mengucapkan terima kasih kepada DRPM Direktorat Jenderal Penguatan Riset dan Pengembangan Ristekdikti yang telah memberikan hibah penelitian. Ucapan terima kasih juga penulis ucapkan kepada rekan dosen Institut Teknologi Adhi Tama Surabaya yang telah banyak membantu penulis dalam menyelesaikan penelitian ini

DAFTAR PUSTAKA

- [1] J. Faber Archila and M. Becker, "Mathematical models and design of an AGV (Automated Guided Vehicle)," in 2013 IEEE 8th Conference on Industrial Electronics and Applications (ICIEA), 2013, pp. 1857–1862.
- [2] X. Zou, D. Huang, D. Feng, and X. Wu, "Design of a double-deck automatic guided vehicle with magnetic traction for material transporting," in 2014 IEEE International Conference on Mechatronics

- and Automation, 2014, pp. 1674–1679.
- [3] Y. M. Abueejela and H. A. Ali, “Wheeled Mobile Robot Obstacle Avoidance Using Compass and Ultrasonic,” *Univers. J. Control Autom.*, vol. 6, no. 1, pp. 13–18, 2018.
 - [4] R. Ismail, Z. Omar, and S. Suaibun, “Obstacle-avoiding robot with IR and PIR motion sensors,” *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 152, no. 1, p. 012064, Oct. 2016.
 - [5] S. Adarsh, S. M. Kaleemuddin, D. Bose, and K. I. Ramachandran, “Performance comparison of Infrared and Ultrasonic sensors for obstacles of different materials in vehicle/ robot navigation applications,” *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 149, no. 1, p. 012141, Sep. 2016.
 - [6] D. Neumann, T. Langner, F. Ulbrich, D. Spitta, and D. Goehring, “Online vehicle detection using Haar-like, LBP and HOG feature based image classifiers with stereo vision preselection,” in *2017 IEEE Intelligent Vehicles Symposium (IV)*, 2017, pp. 773–778.
 - [7] K.-P. Chou et al., “Fast Deformable Model for Pedestrian Detection with Haar-like features,” in *2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, 2017, pp. 1–8.
 - [8] B. Winarno and M. W. P. Pratama, “Pengendalian Gerak Robot Penghinder Halangan Menggunakan Citra dengan Kontrol PID,” *JEECAE (Journal Electr. Electron. Control. Automot. Eng.*, vol. 1, no. 1, Feb. 2017.
 - [9] D. E. J. Sudirman, “Sistem Kontrol Gerak Sederhana Pada Robot Penghinder Halangan Berbasis Kamera Dan Pengolahan Citra,” *SEMNASSTEKNOMEDIA ONLINE*, vol. 5, no. 1, pp. 4-3–31, Feb. 2017.
 - [10] Z. Abidin, K. Joni, and A. F. Ibadillah, “Rancang Bangun Robot Penghinder Halangan Berbasis Kamera Menggunakan Deteksi Kontur,” *J. INFOTEL*, vol. 9, no. 3, pp. 248–256, Aug. 2017.
 - [11] R. A. Firmansyah, D. Purwanto, and R. Mardiyanto, “Pendeteksi Docking Station Menggunakan Rotation Invariant Local Binary Pattern Pada Visual Based Service Robot,” *Semnasteknomedia Online*, vol. 3, no. 1, pp. 3-2–19, Feb. 2015.
 - [12] J. Sankari and R. Imtiaz, “Automated guided vehicle(AGV) for industrial sector,” in *2016 10th International Conference on Intelligent Systems and Control (ISCO)*, 2016, pp. 1–5.